12th International Conference on Information Fusion
Seattle, WA, USA, July 6-9, 2009

Ontological representation of context knowledge for visual data fusion

Juan Gómez-Romero
GIAA, University Carlos III of Madrid
Madrid, Spain
jgromero@inf.uc3m.es

Miguel A. Patricio
GIAA, University Carlos III of Madrid
Madrid, Spain
mpatrici@inf.uc3m.es

Jesús García
GIAA, University Carlos III of Madrid
Madrid, Spain
jgherrer@inf.uc3m.es

José M. Molina
GIAA, University Carlos III of Madrid
Madrid, Spain
molina@ia.uc3m.es

Abstract – *Context knowledge is essential to achieve successful information fusion, especially at high JDL levels. Context can be used to interpret the perceived situation, which is required for accurate assessment. Both types of knowledge, contextual and perceptual, can be represented with formal languages such as ontologies, which support the creation of readable representations and reasoning with them. In this paper, we present an ontology-based model compliant with JDL to represent knowledge in cognitive visual data fusion systems. We depict the use of the model with an example on surveillance. We show that such a model promotes system extensibility and facilitates the incorporation of humans in the fusion loop.*

Keywords: high-level data fusion, computer vision, surveillance systems, ontologies.

1 Introduction

The ultimate objective of a visual fusion system is to detect, identify, and predict the actions that are being performed in the observation area, in order to provide users with knowledge to evaluate threats and to make decisions consequently. Inherent to this cognitive process is the participation of context knowledge. It is accepted that fusion systems, and specifically computer vision systems, must incorporate, either implicitly or explicitly, context knowledge [1, 2]. Interpretation of data and recognition of activities will be hardly successful if contextual information is not considered. Context knowledge constrains the possible interpretations of the perceptions and aids the sensor data to be completed, made sense, and even corrected. Nevertheless, the prevailing JDL model for data fusion considers context knowledge vaguely [3], which has resulted in the proliferation of ad hoc solutions.

An important amount of knowledge in fusion systems must be previously introduced by human analysts. To certain extent, this knowledge can be also considered context, since it is not directly acquired by (visual) sensors. Human entries are not limited to a priori information, and users can continuously provide input to the system to be considered in the fusion process. In this way, soft human entries must be fused with hard sensor data, which causes that humans become an important component of the fusion loop.

The participation of heterogeneous information sources in the fusion process requires the use of a common representation model. Likewise, the fusion system has to make users available suitable tools to interact with it. The focus of data fusion has been mainly low-level sensor fusion, which does not often require sophisticate knowledge representation mechanisms. Recently, this interest is shifting to high-level information fusion, which needs expressive and interpretable representation and reasoning formalisms for situation assessment and impact evaluation. Knowledge management for high-level fusion poses various challenges to the data fusion community [4]: (i) to discern what information should be represented; (ii) to determine which representation formalisms are appropriate; (iii) to elucidate how acquired and a priori information are transformed from numerical measures to symbolic descriptions, according to the JDL levels.

In this paper, we study the use of ontologies to overcome these issues. Ontologies have been recognized as appropriate representation formalisms in information fusion [5] and computer vision [6], since they are formal, extensible, and reusable. Ontologies are defined as highly-expressive, logic-based knowledge models aimed to the description of a domain from a common perspective by using a language that can be processed automatically [7]. This language is usually (equivalent to) a decidable Description Logics [8], e.g. OWL [9].

We propose an epistemological, functional and structural ontology-based model to manage contextual and sensorial data in fusion systems, with a special focus on visual fusion systems. The model identifies: (i) which information is represented in each one of the successive stages that visual signals go through to become decision-support information, in consonance with the JDL layers; (ii) which are the processes that need to be carried out; (iii) how context is represented and applied to accomplish them. Accordingly, the model establishes a set of ontologies to describe the information involved in these processes at each JDL level, and a set of procedures to reason and transform information between them. The ontology-based model can be adapted to different domains, and especially to surveillance and security applications. In this paper, we focus on the description of the overall architecture of the proposal, the structure of the knowledge models, and how they are

applied to support data and information fusion procedures (mainly at the L1 and L2 JDL levels).

The main contribution of this research work is that it provides a theoretical basis for the design of cognitive man-machine vision systems that incorporate context knowledge. We show in detail how the ontological model is materialized, an issue that remains quite unexplored in information fusion. The use of ontologies results in several advantages: (i) abstract representation of fusion information, which improves interpretability of the system and make it easier for the user to interact with it; (ii) reasoning with logic-based formalisms, which allows inferring new knowledge; (iii) extensibility and reusability of the knowledge bases, and application of the model in diverse application domains; (iv) standardization, which facilitates interoperation between different modules and systems.

The remainder of this paper is organized as follows. In Sect. 2, we review some related work pertaining to the incorporation of context knowledge in the JDL model and the use of ontologies for knowledge representation in visual fusion systems. In Sect. 3, we describe the architecture of our approach in relation with the JDL model, paying special attention to the role of the context knowledge. In Sect. 4, we briefly present the ontologies of the model. In Sect. 5, we illustrate the use of the ontology with a practical example on surveillance in secured areas. Finally, the paper concludes with a discussion of the results and plans for future research work.

2 Related Work

Context management and exploitation in fusion systems has not been intensively studied from a general perspective. The last revision of JDL highlights the importance of context knowledge [10], especially when it comes to high-level fusion or improvement of task performance with high-level results, but it is quite unspecific about how it should be acquired, represented, and handled during the process.

A discussion on the role of context knowledge in fusion can be found in [1]. In this research work, the authors discuss several aspects of context representation and stress its applicability in data estimation, association, and alignment. Although ontologies are not studied in detail, they are proposed as a suitable formalism for representing context knowledge. In contrast, classical approaches usually used particular formalisms, which make it difficult to reuse and extend the knowledge bases. For example, in computer vision for surveillance, first order logic-based representations have been considered [2, 11].

Hence, the use of ontologies for data and information fusion in different JDL levels is becoming more and more frequent, as envisioned in [12]. Nevertheless, most of these approaches combine contextual and perceptual information, but do not explicitly describe how context is characterized and integrated in the fusion loop.

In [13], an approach to the development of ontologies for L2 fusion is presented. The authors propose a methodology to create domain-specific ontologies for fusion based on the upper-level ontology BFO (Basic Formal Ontology) and its sub-models SNAP and SPAN (for entities and process, respectively). Other contribution is STO (Situation Theory Ontology), which encodes Barwise's situation semantics [14]. Particularly aimed to computer vision is the research work depicted in [15], which presents a proposal for scene interpretation based on Description Logics and supported by the reasoning features of RACER[1] inference engine. Security applications are studied in [16], which develops an OWL ontology enhanced with rules to represent objects and actors in surveillance systems.

Conversely, other approaches aim at modeling video content at L1. For example, in [17] it is presented a framework for video event representation and annotation. In this framework, VERL (Video Event Representation Language) defines the concepts to describe processes (entities, events, time, composition operations, etc.), and VEML (Video Event Markup Language) is a XML-based vocabulary to markup video sequences (scenes, samples, streams, etc.). VEML 2.0 has been expressed in OWL-DL. Between L0 and L1 can be classified the contribution described in [18], which resembles our idea of creating a symbolic representation of the actual data managed by the tracking algorithms (see Sect. 4.1).

At L0 level, one of the most important contributions is COMM (Core Ontology for MultiMedia) [19], an ontology to represent MPEG-7 data with OWL. COMM does not aim at representing high-level entities, such as people, events, or activities occurring in the scene. Instead, it identifies the components of a MPEG-7 video sequence in order to link them with (Semantic) Web resources. Similarly, the Media Annotations Working Group of the W3C is working in an OWL-based language for adding metadata to Web images and videos [20].

3 Model Description

3.1 Architecture of the Model

The architecture of the ontological model is depicted in Fig. 1. The schema shows the structure of the model associated to the fusion system, in correlation to the successive JDL stages, which range from observed data to decision-ready information. From left to right, visual sensor data is processed by a tracking algorithm, made corresponding to a domain entity, interpreted to recognize the current activity, and evaluated to determine the impact of the threat.

The ontologies of our model can be regarded as vocabularies to express the fusion knowledge at different abstraction levels. From low-level track data to high-level situations, ontologies are used to describe:

[1] http://www.racer-systems.com

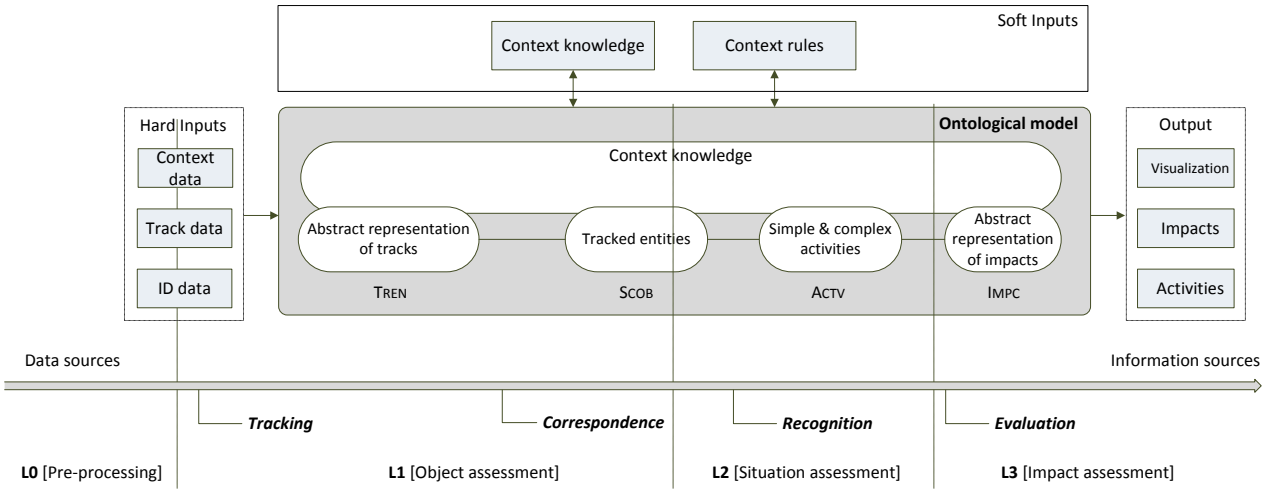

Figure 1. Architecture of the ontology-based visual information fusion model

- *Tracking data (L1)*. Data from the tracking algorithm: tracks and track properties (color, position, velocity, etc.), frames, etc.

- *Scene objects (L1-L2)*. Real-world entities of the scene, properties, and relations: moving objects, actors, topological relations, etc.

- *Activities (L2)*. Behavior descriptions: grouping, approaching, picking/leaving an object, etc.

- *Impacts and threats (L3)*. Association of a cost value to activity descriptions and predictions of future events.

The ontologies contain both context and perceptual data. For instance, the scene objects ontology includes primitives for representing dynamic and static objects. Dynamic object perceptual features (color, position, velocity, etc.) are obtained by the tracking algorithm, whereas the static contextual object features are likely to be previously specified by a human user. In Sect. 4, we analyze how context knowledge is represented in the model and how it is used to improve scene interpretation. It is also interesting to note that an ontology of a higher level includes the ontologies of the lower levels, since the more abstract knowledge is expressed in terms of the less abstract one.

The model has been designed with a view on promoting extensibility and modularity. At each level, it provides a skeleton that includes general concepts and relations to describe the mentioned fusion entities and relations. The developer must refine this vocabulary and extend the ontologies according with her objectives. In this manner, we clearly differentiate between general and domain-specific knowledge, which is essential to guarantee the applicability of the approach in assorted application areas. Accordingly, the contents of the ontologies of the model are a tradeoff between generality and utility. Each one of them has to be general to be used in different domains, but also it has to include as much description terms as possible to make it easy to extend it in each case

As we explain in the following section, the ontologies of the model at lower abstraction levels are larger than ontologies at higher levels. This does not mean that in final systems less abstract knowledge is prevalent, but that our model, which has to be extended, provides fewer constructors at high levels. At low level, more knowledge is common to different applications. For instance, tracks are managed in every fusion system, and therefore the track description ontology of the model has to be scarcely extended. On the contrary, high-level ontologies describing activities are very domain-specific, and therefore, the one of the model will have to be more extensively completed.

In the model, we distinguish two types of reasoning processes. Firstly, reasoning procedures can be applied to infer additional knowledge from the explicit facts within each ontology. By using a Description Logics inference engine (e.g. RACER), different standard reasoning tasks can be performed. For instance, all concept inclusions (asserted and deduced) can be computed, which is known as ontology classification. It is as well possible to perform other non-standard inferences or even add rules to increase the expressivity of the knowledge bases. These are cases of deductive reasoning, since they take a set of facts as the input and apply logical resolution to compute derived information.

The second reasoning task concerns the transformation of numeric data to symbolic objects, where the global input of the process is the video sequence and the tracks, and

the final output are the activities and impacts associated to the scene. In other words, it is necessary not only to reason within each stage, but also to transform lower to higher level data between stages. This can be regarded as a type of abductive reasoning, in contrast to the deductive reasoning performed in the former case. Abductive reasoning takes a set of facts as the input and finds a suitable hypothesis that explains them. In our model, determining if a track corresponds to a person or to a moving object can be regarded as this type of reasoning. Therefore, abductive reasoning processes must be performed to convert knowledge expressed in an ontology to knowledge expressed in a higher level ontology. Abductive reasoning is out of the scope of classical Description Logics [21], but in our case, it can be simulated by using customized procedures or, more interestingly, by defining transformation rules (Sect. 5).

In the remainder of this section, we discuss the role of context knowledge in our model for visual data fusion. In the next section, we describe in more detail the ontologies that participate in it.

3.2 Context Knowledge

There is not a consensus about what should be considered context in fusion and vision systems. A traditional definition of context, appeared in the area of Ubiquitous Computing, establishes that context is any information (either implicit or explicit) that can be used to characterize the situation of an entity [22]. Other definitions have been expressly proposed for computer vision [2], most of them focusing on the distinction between the perceived stimulus and the outer information that affect their comprehension. These authors highlight that context includes information about the scene environment, information about the parameters of the recording, information previously computed by the vision system, and user-requested information.

In accordance to these approaches, we consider that context is all the *additional information* about the interesting entities of the scene. By additional, we exclude all the data that can be automatically extracted from the scene. Insofar as visual information fusion is concerned, context is external knowledge used to complete the purely quantitative interpretation of a scene that is performed by image analysis algorithms. For instance, we do not consider track properties (obtained by the tracking algorithm) as context information. Conversely, static objects features, such as motionless object size, position, occlusion, etc., are regarded as context information. Time and location of the scene, acquired from sensors or introduced by the user, are also considered context information. Another example of context is a rule stating that the density of tracks in a frame is higher during rush hours. In any case, the delimitation between contextual and non-contextual knowledge is not exhaustive and can be adapted to the requirements of each application, without prejudice to the generality of our approach.

As a result of this definition of context, the role of the human analysts is crucial. A large amount of context knowledge is expected to be provided by these users, and the cognitive ability of the system strongly depends on it. Part of the context is a priori and common-sense knowledge that is introduced before the initialization of the system. For instance, a region of the video can be marked as a door, which means that tracked entities go in and out of the scene through it. Part of the context is learnt during the execution of the system. For instance, if tracks are created or removed more frequently when they enter a region of the image, it can be supposed that this region is a door. In both situations, participation of the user is useful (required in the former case, desirable in the latter one). As emphasized in the introduction, users must be provided with usable presentation and control interfaces. The ontology-based model describes abstractly the system information, in such a way that it is more easily interpretable, and therefore interaction procedures can be implemented straightforwardly.

Context knowledge spans through all the levels of the JDL model. Context knowledge may be physical, logical, or cognitive; static or dynamic; general or specific; descriptive or deductive; etc. This means that we have sub-context models at L1, L2, etc. Thus, context is included at each level along with acquired knowledge, sometimes even indistinguishably. It can be considered however that a global context model exists and it encompasses all the contextual knowledge embedded in the tracks, entities, activities, and impact sub-ontologies. The distinction can be made explicit if a different namespace is associated to context knowledge. Context knowledge is used in the two types of reasoning procedures described in the previous section. Along with perceptual knowledge, it can be applied to enhance deduction within a level and abduction between levels.

Necessarily, the granularity and the amount of context knowledge managed by the fusion system determine the possible interpretation of the scenes. For example, if only L1 context is considered, scene interpretation will be restricted to individual object properties, and interactions between them will not be analyzed. If L2 context is considered, the scene will be interpreted in terms of object properties and relations between objects. Clearly, the incorporation of more context knowledge results in better recognition of the situations.

4 Model Ontologies

An excerpt of the model ontologies is presented in Fig. 2. This figure depicts the layered structure of the model, as well as the distinction between general (i.e. in the ontologies of the model) and domain-specific (i.e. to be created by developers) knowledge.

Fig. 2 shows some classes of the ontologies, the relations between them, and restrictions to these relations. Concepts in grey are the entities that link the representations at

different levels. For instance, SceneObject, which is a L1 concept, is imported by L2 ontologies, which describe activities from object interactions.

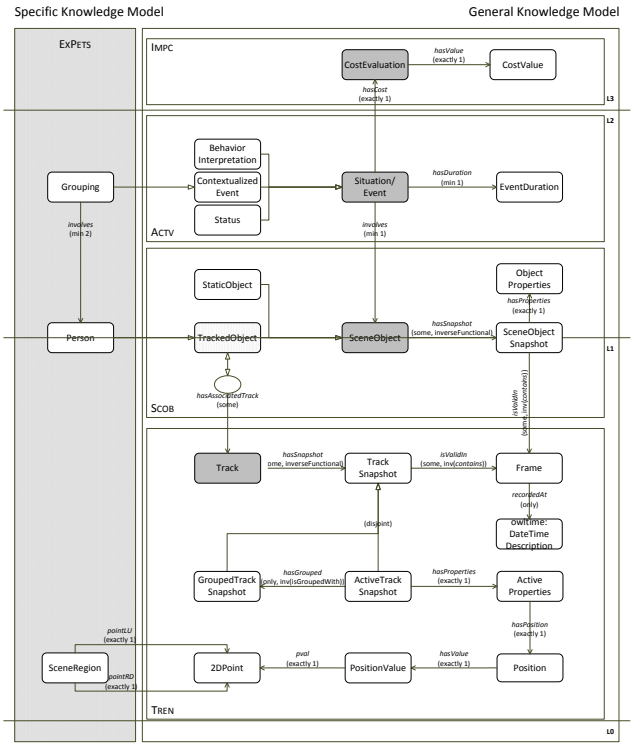

Figure 2. Model ontologies structure

4.1 Object Assessment Knowledge

The L1 ontologies represent tracks and tracked entities information. We have separated these two types of knowledge in the tracking entities (TREN) and the scene objects (SCOB) description ontologies, respectively.

4.1.1 Representation of Tracking Data

The core concepts in TREN are Frame and Track. A frame is identified by an ID and is marked with a time stamp (using OWL-Time [23]). The definition of tracks is more complex. It is necessary to design an ontology that can represent the temporal evolution of the scene, and not only its state in a given instant. That is, we want to keep all the information related to a track during the complete sequence (activity, occlusions, position, size, velocity, etc.), which changes between frames, and not only its lastly updated values. Therefore, we must connect tracks, frames, and track properties at each frame, which is a ternary relation. Furthermore, track features must be defined as general as possible, in such a way that they can be extended.

To solve the first issue, we have followed a design pattern proposed by the W3C Semantic Web Best Practices and Deployment Working Group to define ternary relations in OWL ontologies [24]. We have associated a set of TrackSnapshots to each Track. Each TrackSnapshot, with property values, is asserted to be valid in various Frames. To solve the second issue, we have followed the *qualia* approach, used in the upper ontology DOLCE [25]. This pattern distinguishes between properties themselves and the space in which they take values. For instance, we have defined a Position concept that is related with a positionValue property to a value of the PositionValue space.

Since the TREN ontology is specifically intended to describe the data provided by the tracking algorithm, our definition of context is hardly applicable at this level. Nevertheless, additional axioms or rules to calculate complex properties of tracks (e.g. distances), as well as spatial relationships (inclusion, adjacency, etc.), could be regarded and created as TREN context.

4.1.2 Representation of Scene Objects

Scene objects are real-world entities that have a visual materialization. A general representation for these objects and their properties to be extended in particular applications is defined in the SCOB ontology. For example, in surveillance applications, concepts such as person, door, or column, will be created to extend the more general SCOB concepts. SCOB mainly contains L1 knowledge, that is, knowledge about single object without considering interactions between them. However, it may be interesting to represent some relations between objects not strictly pertaining to activities. For this reason, the tracked entities knowledge and the associated context knowledge is halfway between L1 and L2.

The main concept in SCOB is SceneObject. SceneObject is an abstract class that includes all the interesting objects in the scene, either dynamic (i.e. TrackedObject) or contextual (i.e. StaticObject). SceneObjects have properties, e.g. position, illumination, behavior, etc. Properties are also represented in terms of snapshots and *quales*, in the same manner as it has been explained for TrackSnapshots. In this way, it is possible to describe the properties of an object during its whole life and to add new properties easily.

An abductive reasoning procedure to calculate the correspondence between graphical tracks (TREN instances) and real-world objects (SCOB instances), i.e. to determine 'correspondence' between observed tracks and expected objects, must be developed. The implementation still remains to the application developer, but with the advantage that it can rely on the ontological model. With context and sensor data described formally, semantic procedures can be created without effort. In the next section we provide an example of abductive reasoning for achieving correspondence by using rules.

4.2 Representation of Activities and Impacts

The ACTV ontology provides a vocabulary for describing scene activities. Activities are defined in terms of relations

between scene objects expressed in the SCOB ontology. That means that the ACTV ontology imports SCOB. Likewise, the IMPC ontology, which contains terms to describe activity impacts, is built on top of ACTV.

Since the number of possible scenes is countless, only very general activities have been defined in ACTV. Domain-specific activities must be created by refinement of the elements of the ACTV ontology (e.g. Grouping). In this approach to the problem, we have reused part of the formulation of the ontology presented in [26]. We have also introduced some properties to establish the temporal duration of the activities. The IMPC ontology, in turn, contains a vocabulary to associate an evaluation value to ACTV activities. This value can be a simple numerical assessment or, more probably, a complex expression suggesting or predicting future actions.

At these levels, the difference between contextual and sensorial knowledge is practically inexistent. It is not possible to state that an activity is exclusively perceived or contextual, since its expression relates static and dynamic objects. Furthermore, activities are recognized by taking into account a considerable amount of external and a priori knowledge. This means that it is not possible to make a distinction, and that context is embedded at the L2 level. This argument is directly applicable to L3.

In accordance, it is important to stress once again that the ontology can be only used for describing activities and inferring implicit actions from the explicit assertions. Interpreting a scene ('recognition') based upon scene object properties requires abductive reasoning. The advantage is that objects and activities are formally characterized with the SCOB and ACTV ontologies, which assist this procedure. These considerations are also valid for the evolution between ACTV and IMPC ('evaluation').

5 Example

To exemplify the creation of and the reasoning with scene descriptions, we have used a sequence of the PETS dataset[2]. In this recording, various people walk in front of a shop window. We have developed a specific EXPETS ontology that adapts the generic model presented in the previous section to this scenario by adding concepts such as person, door, or column. The PETS video is processed by a tracking algorithm and then, the result data is inserted as ontology instances. In the first example, we show how a track is described with the TREN ontology and how knowledge is used to assign the track to a predefined type of object. In the second example, we show how activities can be recognized by applying abduction rules.

[2] The Performance Evaluation of Tracking and Surveillance (PETS) dataset has available numerous scenarios. We have chosen a minute-long sequence from the PETS2002 workshop, where the underlying task was to track pedestrians in indoor video sequences of a shopping mall (http://www.cvg.cs.rdg.ac.uk/PETS2002/pets2002-db.html).

Figure 3. Person in PETS sequence

Example 1. Fig. 3 shows a frame of the sequence. A person is moving in the scene, which is detected by the underlying video tracking algorithm. An excerpt of the description in OWL Manchester syntax [27] of the track instance detected by the algorithm using TREN (the L1 ontology) concepts and relations is shown below. Instances are marked in italics, and concepts are underlined.

```
Individual: track1
  Types: Track
  Facts:
    hasSnapshot: tr1sn1

Individual: tr1sn1
  Types: TrackSnapshot
  Facts:
    isValidIn   frame2
    hasActualProperties prop1

Individual: prop1
  Types: ActiveProperties
  Facts:
    hasPosition pos1
    hasSize siz1

Individual: pos1
  Types: Position
  Facts:
    hasValue posval1

Individual: posval1
  Types: PositionValue
  Facts:
    pval point1

Individual: point1
  Types: 2DPoint
  Facts:
    x 250
    y 100
```

Abductive *if-then* rules expressing a priori contextual knowledge can be defined to create objects to associate tracks to, that is, to transform L1 knowledge into L2 knowledge. For example, the following rule infers that, given the size and the region where is located a track snapshot which has not been identified, it can be guessed that it is a person. The rule is written in RACER rule language. We assume that suitable implementation for new predicates (dimensions, inclusion, etc.), marked in bolds, have been also developed. Terms preceded by '?' are variables, and terms in italics are constants. Concept and property predicates are show in roman.

```
Track(?t) ^
TrackSnapshot(?tsn) ^
hasSnapshot(?t, ?tsn) ^
isValidIn(?tsn, currentFrame) ^
not(hasAssociatedTrack(?an_obj, ?t)) ^
inside(?tsn, region1) ^
width(?tsn, ?w) ^ height(?tsn, ?h) ^
greaterThan(?w, l1) ^ greaterThan(?h, l2)
--->
Person(?p) ^
hasAssociatedTrack(?p, ?t) ^
ObjectSnapshot(?new_osn) ^
hasSnapshot(?p,?new_osn) ^
isValidIn(?new_osn, currentFrame)
```

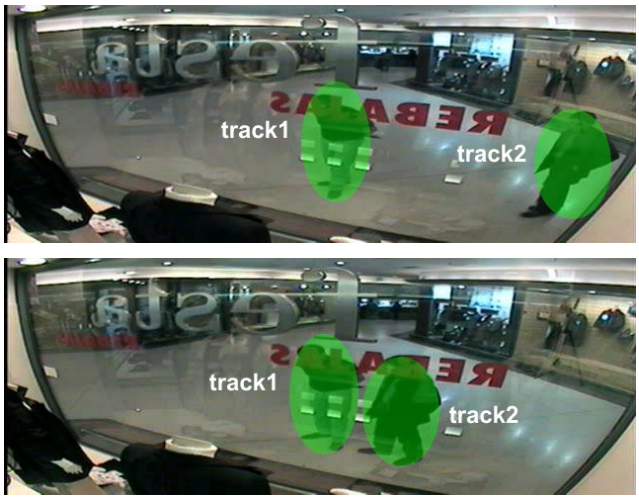

Figure 4. People grouping in PETS sequence

Example 2. Let us suppose now the situation of Fig. 4, which shows two persons grouping. A rule such as the following could be defined to infer the higher level activity from the lower-level object descriptions: The following rule establishes that if the distance between two persons, with associated tracks, is reduced during a predetermined number of successive frames, they are grouping.

```
Person(?p1) ^ Person(?p2) ^
hasSnapshot(?p1, ?psn1) ^
hasSnapshot(?p2, ?psn2) ^
isValidIn(?psn1, currentFrame) ^
isValidIn(?psn2, currentFrame) ^
distance(?psn1, ?psn2,?d)) ^
hasSnapshot(?p1, ?psn1prev_1) ^
hasSnapshot(?p2, ?psn2prev_1) ^
isValidIn(?psn1prev_1, prevFrame_1) ^
isValidIn(?psn2prev_1, prevFrame_1) ^
distance(?psn1, ?psn2, ?dprev_1)) ^
…
lessThan(?d,?dprev_1) ^…^ lessThan(?…,?dprev_n)
--->
Grouping(?act) ^
involves(?act, ?p1) ^ involves(?act, ?p2) ^
hasDuration(?act, ?duration) ^
begins(?duration, prevFrame_n) ^
ends(?duration, currentFrame) ^
```

This example is quite simple, but it can be easily seen that the procedure can be extended without difficulty to more complex object interactions.

6 Conclusions and Future Work

In this paper, we have proposed a context-based model to support information and data fusion in computer vision. The model encompasses a set of ontologies that are used to describe the sensorial and contextual knowledge of the system. We have studied the information that should be considered at each JDL level and we have created general ontologies to represent it. General ontologies are specialized in each application domain, offering a common and reusable framework for the development of visual data and information fusion systems. The model explicitly considers context knowledge, which is a key factor to accomplish fusion objectives, and provides suitable mechanisms to represent it.

Formal representation of knowledge in fusion has several advantages. The symbolic model of the scene is more interpretable, which facilitates the incorporation of the human analyst in the fusion loop. It is also possible to reason with them in order to: (i) deduce implicit knowledge from the explicit descriptions; (ii) infer explanations for the observed facts with the aim of creating more abstract representations. The use of ontologies makes it easy to extend the knowledge bases and to interoperate between components and with other systems. Interestingly enough, our representation allows the description of the temporal evolution of the system, and not only its state in a precise instant.

We plan to continue this research work various directions. First, we will fully integrate the ontological representation with our tracking software [28]. This may imply further refinements or simplifications of the current model, which has been developed with a very broad scope. To test the solution in real domains, suitable descriptive ontologies extending the model and abduction rules will have to be created (manually or semi-automatically), which poses a serious challenge because it may require a considerable effort. External context data sources (e.g. weather) will be also considered, which will require implementing appropriate middleware to incorporate them into the model. Moreover, we will study how reasoning within the model could be applied to provide feedback to the tracking system, i.e. how to modify the behavior of the algorithm according to the scene and the context. We strongly believe that the use of our formal knowledge representation model will result in a significant improvement of the computer vision system, which will be able to understand more aspects of the scene and apply this knowledge to enhance image-processing algorithms.

Acknowledgement

This work was supported in part by Projects CICYT TIN2008-06742-C02-02/TSI, CICYT TEC2008-06732-C02-02/TEC, SINPROB, CAM MADRINET S-0505/TIC/0255 and DPS2008-07029-C02-02.

References

[1] A.N. Steinberg, and G. Rogova, "Situation and context in data fusion and natural language understanding," *11th International Conference on Information Fusion*, Cologne, Germany, 2008, pp. 1-8.

[2] F. Bremond, and M. Thonnat, "A context representation for surveillance systems," *ECCV Workshop on Conceptual Descriptions from Images*, Cambridge, UK, 1996.

[3] A.N. Steinberg, and C.L. Bowman, "Rethinking the JDL data fusion levels," *MSS National Symposium on Sensor and Data Fusion*, Columbia, SC, USA, 2004.

[4] D. Lambert, "Grand challenges of information fusion," *6th International Conference on Information Fusion*, Cairns, Australia, 2003, pp. 213-220.

[5] J. Llinas, C. Bowman, G. Rogova, A. Steinberg, E. Waltz, and F. White, "Revisiting the JDL data fusion model II," *7th International Conference on Information Fusion*, Stockholm, Sweden, 2004, pp. 1218-1230.

[6] N. Maillot, M. Thonnat, and A. Boucher, "Towards ontology-based cognitive vision," Machine Vision and Applications, Vol. 16, pp. 33-40. 2004.

[7] R. Studer, V.R. Benjamins, and D. Fensel, "Knowledge Engineering: Principles and Methods," *Data Knowledge and Engineering*, Vol. 25, pp. 161-197. 1999.

[8] F. Baader, I. Horrocks, and U. Sattler, "Description Logics as Ontology Languages for the Semantic Web," *Mechanizing Mathematical Reasoning*, Springer Berlin / Heidelberg, pp. 228-248. 2005.

[9] D.L. McGuinness, and F. van Harmelen, "OWL Web Ontology Language Overview," World Wide Web Consortium Recommendation, 2004. (Available at: http://www.w3.org/TR/owl-features/).

[10] A.N. Steinberg, and C.L. Bowman, "Revisions to the JDL Data Fusion Model," *Handbook of Multisensor Data Fusion*, M.E. Liggins, D.L. Hall, and J. Llinas, eds., CRC Press, pp. 45-67. 2009.

[11] O. Brdiczka, P.C. Yuen, S. Zaidenberg, P. Reignier, and J.L. Crowley, "Automatic acquisition of context models and its application to video surveillance," *18th International Conference on Pattern Recognition*, Hong Kong, 2006, pp. 1175-1178.

[12] C. Nowak, "On ontologies for high-level information fusion," *6th International Conference on Information Fusion*, Cairns, Australia, 2003, pp. 657-664.

[13] E.G. Little, and G.L. Rogova, "Designing ontologies for higher level fusion," *Information Fusion*, Vol. 10, pp. 70-82. 2009.

[14] M.M. Kokar, C.J. Matheus, and K. Baclawskim, "Ontology-based situation awareness," *Information Fusion*, Vol. 10(1), pp. 83-98. 2009.

[15] B. Neumann, and R. Möller, "On scene interpretation with Description Logics," *Image and Vision Computing*, Vol. 26, pp. 82-101. 2008.

[16] L. Snidaro, M. Belluz, and G.L. Foresti, "Domain knowledge for surveillance applications," *10th International Conference on Information Fusion*, Quebec, Canada, 2007, pp. 1-6.

[17] A.R. François, R. Nevatia, J. Hobbs, R.C. Bolles, and J.R. Smith, "VERL: an ontology framework for representing and annotating video events", *IEEE Multimedia*, Vol. 12 (4), pp. 76-86. 2005.

[18] M. Kokar, and J. Wang, "Using ontologies for recognition: an example," *5th International Conference on Information Fusion*, Annapolis, MD, USA, 2002, pp. 1324-1330.

[19] R. Arndt, R. Troncy, S. Staab, L. Hardman, and M. Vacura, "COMM: designing a well-founded multimedia ontology for the web", *6th International Semantic Web Conference*, Busan, South Korea, 2008, pp. 30-43.

[20] W. Lee, T. Bürger, and F. Sasaki, "Use cases and requirements for ontology and API for media object 1.0", 2009. (Available at: http://www.w3.org/TR/media-annot-reqs/).

[21] C. Elsenbroich, O. Kutz, and U. Sattler, "A case for abductive reasoning over ontologies," *OWL: Experiences and Directions Workshop*, Athens, Georgia, USA, 2006.

[22] A. Dey, G. Abowd. "Towards a Better Understanding of Context and Context-Awareness," *CHI Workshop on the What, Who, Where, When, and How of Context-Awareness*, The Hague, Netherlands, 2000.

[23] J. Hobbs, and F. Pan, "Time ontology in OWL," 2006. (Available at: http://www.w3.org/TR/owl-time/).

[24] N. Noy, and A. Rector, "Defining n-ary relations on the Semantic Web," 2006. (Available at: http://www.w3.org/TR/swbp-n-aryRelations/).

[25] A. Gangemi, N. Guarino, C. Masolo, A. Oltramari, and L. Schneider, "Sweetening Ontologies with DOLCE," *13th International Conference on Knowledge Eng. & Knowledge Man.*, Sigüenza, Spain, 2002, pp. 223-233.

[26] C. Fernández, and J. González, "Ontology for Semantic Integration in a Cognitive Surveillance System," 2nd International Conference on Semantic and Digital Media Technologies, Genoa, Italy, 2007, pp. 260-263.

[27] M. Horridge, N. Drummond, J. Goodwin, A. Rector, R. Stevens, and H.H. Wang, "The Manchester OWL Syntax," *OWL Experiences and Directions Workshop (OWLEd '06)*, Athens, Georgia, USA, 2006.

[28] M.A. Patricio, F. Castanedo, A. Berlanga, Ó. Pérez, J. García, and J.M. Molina, "Computational Intelligence in Visual Sensor Networks: Improving Video Processing Systems," *Computational Intelligence in Multimedia Processing: Recent Advances*, Springer, pp. 351-377. 2008.